

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И СИСТЕМ

Александровская Людмила Игоревна

Выпускная квалификационная работа бакалавра

**Динамика и процессы управления для подводного
робота-манипулятора**

Направление 010300

"Фундаментальные информатика и информационные технологии"

Научный руководитель,
доктор физ.-мат. наук,
профессор
Веремей Е. И.

Санкт-Петербург

2016

Содержание

Содержание	2
Введение	3
Постановка задачи	5
Обзор литературы	7
Глава 1. Математическая модель подводного робота-манипулятора	9
1.1. Уравнения движения подводного робота-манипулятора	9
1.2. Линеаризация уравнений динамики	13
Глава 2. Управление роботом-манипулятором в вертикальной плоскости	16
2.1. Построение оптимального асимптотического наблюдателя	16
2.2. Построение стабилизирующего закона управления	17
Глава 3. Имитационный моделирующий комплекс в среде Simulink	19
3.1. Компьютерная модель подводного робота-манипулятора	19
3.2. Компьютерная модель асимптотического наблюдателя	21
3.3. Компьютерная модель системы стабилизирующего управления	22
Глава 4. Результаты имитационного моделирования	24
4.1. Анализ работы асимптотического наблюдателя и системы управления	24
Выводы	31
Заключение	32
Список литературы	33
Приложение А	35

Введение

Робот (от чеш. robota – тяжёлый труд, каторга) – автоматическое устройство, созданное для того, чтобы заменить человека в опасных ситуациях и средах и освободить его от однообразных рутинных операций. Современные роботы, созданные на базе последних достижений науки и техники, применяются во всех сферах человеческой жизни [4].

Робототехника – прикладная наука, объединяющая такие дисциплины, как механика, телемеханика, электроника, информатика, электротехника и радиотехника, искусственный интеллект и производство.

Робототехнические комплексы являются основой автоматизации промышленности, медицины и космонавтики. Создание таких комплексов в настоящее время является актуальной задачей, требующей широкого привлечения информационных технологий и вычислительной техники.

Подводная робототехника является одной из новейших областей в рамках данного направления. В последние годы необитаемые подводные аппараты (НПА), оснащенные многозвенными роботами-манипуляторами, широко используются для решения таких задач, как исследование морского дна, мониторинг и обслуживание трубопроводов и кабельных линий, сбор биологических и геологических проб жидкостей, освоение морских нефтяных и газовых месторождений, проведение подводных аварийно-спасательных работ. Объясняется это тем, что применение пилотируемых аппаратов очень опасно и несет большие эксплуатационные расходы.

Исследования в области моделирования и управления подводными роботами в конечном итоге направлены на достижение полной автоматизации указанных процессов. Данная цель является исключительно сложной задачей, так как подразумевает решение широкого спектра теоретических и практических проблем, подлежащих всесторонним исследованиям.

В частности, система датчиков НПА имеет дело с неизвестной и плохо структурированной окружающей средой, не допускающей применение системы GPS (Global Positioning System) в связи с невозможностью передачи электромагнитных волн определенных частот. Применение систем, основанных на компьютерном зрении, не обеспечивает достаточной надежности при наличии плохой видимости под водой [9].

Постановка задачи

Подводный манипулятор (ПМ) состоит из тех же частей, что и робот, применяемый в обычных земных условиях, и представляет собой определенную последовательность звеньев, соединенных между собой шарнирами (рис. 1). У каждого манипулятора есть звено, на которое устанавливается прибор, исполняющее определенное действие. Такое звено называется исполнительным.

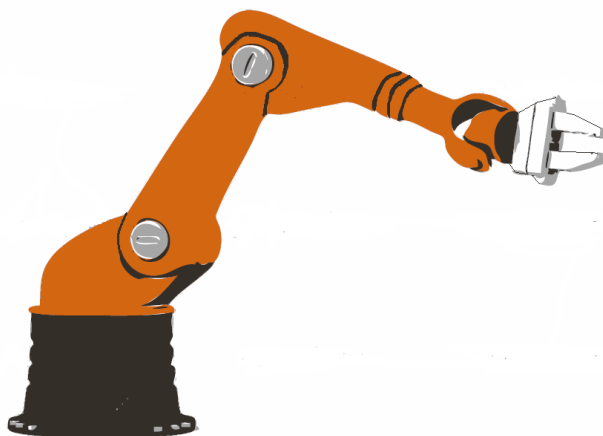


Рис. 1. Промышленный робот-манипулятор.

Соединения между звеньями могут быть подвижными и неподвижными. Подвижные соединения могут обеспечивать возвратно-поступательное движение, вращение вокруг одной оси или вращение вокруг трех осей (шаровой шарнир).

Механизм приводится в действие посредством приводов, которые могут располагаться на основании манипулятора (место закрепления первого звена) или между звеньями.

Каждый манипулятор характеризуется таким понятием, как рабочая зона – пространство, в котором может передвигаться исполнительный орган. Это перемещение может осуществляться как в одной, так и в нескольких плоскостях. Рабочая зона манипулятора определяется длиной звеньев, типом и количеством шарниров и приводов.

Данный ПМ существенно воздействует на подводный аппарат, на котором он установлен: на НПА действуют не только инерционные и гравитационные силы и моменты, но и силы, определяемые движением манипулятора в вязкой окружающей среде, которые приводят к смещению аппарата относительно заданного положения. Это служит дополнительной помехой для решения соответствующих функциональных задач.

В данной работе рассматривается НПА NPS AUV II, разработанный в Naval Postgraduate School (рис. 2) [11], [12]. На этом аппарате установлен робот, представляющий собой двухзвенный манипулятор с вращательными соединениями между звеньями, неподвижно соединенный с НПА. ПМ установлен под подводным аппаратом на середине его длины.

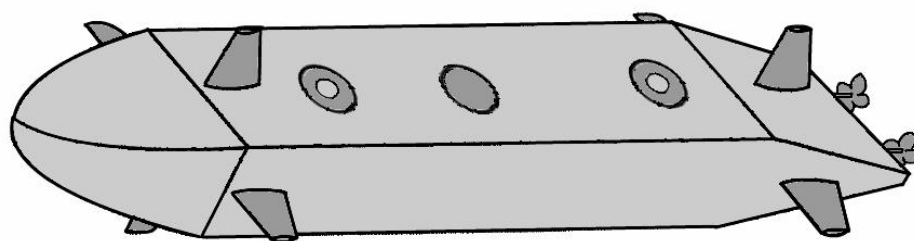


Рис. 2. Схема необитаемого подводного аппарата NPS AUV II.

В данной работе рассматриваются следующие вопросы:

- построение математической модели динамики подводного робота-манипулятора и выполняется её линеаризация;
- формирование оптимального асимптотического наблюдателя состояния объекта;
- синтез эффективного стабилизирующего управления режимом зависания подводного робота в окрестности заданного положения при проведении подводных работ;
- разработка компьютерной модели в среде MATLAB-Simulink;
- исследование поведения замкнутой системы при наличии внешнего возмущения.

Обзор литературы

Решение задач управления динамическими объектами и, в частности, подводными роботами-манипуляторами начинается с построения их математических моделей. Моделирование автономных подводных аппаратов подробно рассматривается в работе [11].

При описании кинематики манипуляторов используют методы, описанные в [3, 6], отличающиеся выбором локальных систем координат, при этом наиболее популярный подход основывается на использовании систем координат Денавита-Хартенберга [5].

При моделировании динамики робота-манипулятора используется стандартный подход, при котором манипулятор рассматривается как последовательная совокупность влияющих друг на друга твердых тел. При этом для расчета гидродинамических сил может использоваться итеративный алгоритм Ньютона-Эйлера [16] или алгоритм, представленный в [13].

Что касается построения законов стабилизирующего управления, то спектр используемых методов гораздо шире, чем для моделирования. В частности, управление необитаемыми подводными аппаратами рассматривается в монографии [11].

Методы синтеза алгоритмов управления подводными роботами направлены главным образом на обеспечение адаптивности и робастности замкнутых систем [9]. Широко используются методы, основанные на принципах теории нечетких множеств [10] и нейронных сетей [14].

В работе [15] проведено численное моделирование и показано, что использование централизованного управления лучше, чем применение независимых управлений (для судна и для манипулятора в отдельности). При этом в работе обосновано увеличение производительности системы при централизованном подходе.

Серьезной проблемой является незавершенность теоретических исследований в области синтеза систем управления для нелинейных объектов. Существующие решения применяются только для ограниченного класса объектов и не носят универсальный характер, при этом сопровождаются значительными вычислительными трудностями. В связи с этим совершенствование и исследование методов синтеза управления нелинейными объектами является актуальной задачей.

Глава 1. Математическая модель подводного робота-манипулятора

Рассматриваемый подводный манипулятор SMART-3S, произведенный компанией COMAU [9], состоит из двух звеньев, которые приводятся в движение электрическими приводами, установленными в соединениях, и имеет две степени свободы. ПМ закреплен на неподвижном основании под судном на середине его длины.

В данной главе представлены уравнения движения объекта управления и рассмотрен вопрос их линеаризации.

1.1. Уравнения движения подводного робота-манипулятора

В работе движение подводного робота рассматривается только в вертикальной плоскости, то есть не происходит бокового движения, движений по курсу и крену. Такое допущение позволяет рассматривать упрощенную модель лодки, которая содержит 3 степени свободы, что в свою очередь упрощает вычисления, а также делает результаты экспериментов более наглядными для анализа.

Поведение объекта управления описываются в ортонормированных системах координат (рис. 3). Положение НПА в базовой земной ($O_e x_e y_e$) системе координат определяются линейными смещениями x , y и значением угла дифферента θ .

Положение манипулятора определяется двумя углами (рис. 4): углом отклонения первого звена от вертикального положения q_1 и углом отклонения второго звена от оси первого звена q_2 . Для каждого звена манипулятора на оси его сочленения формируются системы координат, которые позволяют последовательно преобразовывать координаты захватывающего устройства из системы отсчета, связанной с последним

звеном, в земную систему координат. Нулевой репер располагается в основании судна и является неподвижным по отношению к нему, далее реперы нумеруются по порядку следования звеньев: i -ый репер находится на конце i -ого звена.

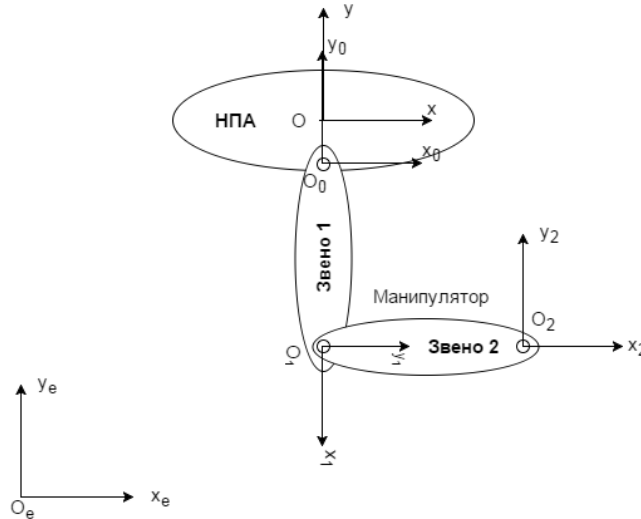


Рис. 3. Системы координат, связанные с землей, судном и звеньями манипулятора.

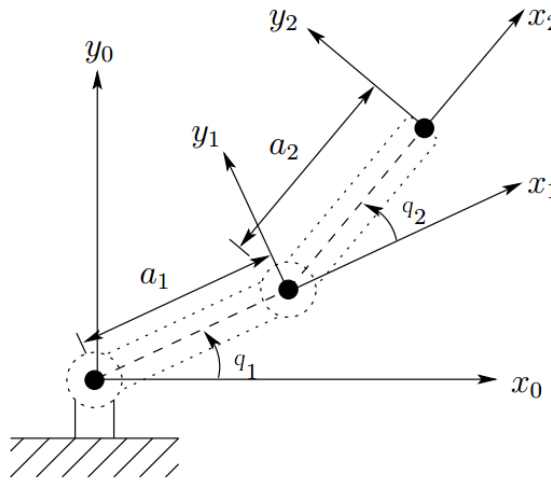


Рис. 4. Формирование координат манипулятора.

Уравнения движения подводного аппарата NPS AUV II подробно рассматриваются в работах [11],[12] и имеют вид:

$$\mathbf{M}_r \dot{\mathbf{v}} + \mathbf{C}_r \mathbf{v}(\mathbf{v}) + \mathbf{D}_r(\mathbf{v}) \mathbf{v} + \mathbf{g}_r(\boldsymbol{\eta}) = \boldsymbol{\tau}_r. \quad (1.1)$$

Здесь \mathbf{M}_r – матрица инерции, включающая присоединенные массы, \mathbf{C}_r – матрица кориолисовых и центробежных сил, \mathbf{D}_r – матрица

демпфирующих гидродинамических сил, \mathbf{g}_r – вектор гравитационных, архимедовых и восстанавливающих сил, $\mathbf{\tau}_r$ – вектор возмущающих и управляющих сил, $\boldsymbol{\eta} = (x, y, \theta)$ – вектор координат НПА в базовой земной системе координат, $\mathbf{v} = (u, w, q)$ – линейные и угловая скорости НПА в связанной системе координат. При этом в данной работе не рассматриваются уравнения бокового движения (по курсу, крену и боковому сносу).

Переход между базовой земной системой координат и связанной системой координат (Oxy) задается уравнением (1.2). Соотношение (1.3) выражает зависимость скоростей устройства захвата $\dot{\mathbf{x}}_e$ от скоростей НПА и обобщенных скоростей манипулятора.

$$\dot{\boldsymbol{\eta}} = \mathbf{J}_R(\boldsymbol{\eta})\mathbf{v}, \quad (1.2)$$

$$\dot{\mathbf{x}}_e = \mathbf{J}_1(\boldsymbol{\eta})\mathbf{v} + \mathbf{J}_2(\mathbf{q})\dot{\mathbf{q}}. \quad (1.3)$$

где $\mathbf{q} = (q_1, q_2)$ – обобщенные координаты манипулятора, $\mathbf{J}_R, \mathbf{J}_1, \mathbf{J}_2$ – матрицы соответствующих размерностей, рассмотренные в работах [11], [16].

Зная скорости \mathbf{v} и ускорения $\dot{\mathbf{v}}$ подводного аппарата, а также скорости $\dot{\mathbf{q}}, \ddot{\mathbf{q}}$ сочленений манипулятора, для каждого звена можно вычислить следующие параметры (верхний индекс означает номер системы координат, в которой выражен параметр):

ω_i – угловую скорость i -ой системы координат,

$\dot{\omega}_i$ – угловое ускорение i -ой системы координат,

\mathbf{v}_i^i – линейную скорость центра i -ой системы координат,

\mathbf{v}_{ic}^i – линейную скорость центра масс звена i ,

\mathbf{a}_i^i – линейное ускорение центра i -ой системы координат.

В связи с тем предположением, что движение происходит только в одной плоскости, приведенные в [9], [16] выражения для этих параметров можно упростить. Приведем их окончательный вид:

$$\omega_{i+1}^{i+1} = \omega_i^i + \dot{q}_{i+1},$$

$$\dot{\omega}_{i+1}^{i+1} = \dot{\omega}_i^i + \ddot{q}_{i+1},$$

$$\mathbf{v}_{i+1}^{i+1} = \mathbf{R}_{i+1}^i \mathbf{v}_i^i + \omega_{i+1}^{i+1} \mathbf{S} \mathbf{d}_{i+1/i}^{i+1},$$

$$\mathbf{v}_{i+1c}^{i+1} = \mathbf{R}_{i+1}^i \mathbf{v}_i^i + \omega_{i+1}^{i+1} \mathbf{S} \mathbf{d}_{i/i_c}^{i+1},$$

$$\mathbf{a}_{i+1}^{i+1} = \mathbf{R}_{i+1}^i \mathbf{a}_i^i + \dot{\omega}_{i+1}^{i+1} \mathbf{S} \mathbf{d}_{i+1/i}^{i+1} - (\omega_{i+1}^{i+1})^2 \mathbf{d}_{i+1/i}^{i+1},$$

где

$$\mathbf{S} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

Здесь \mathbf{R}_{i+1}^i – матрица поворота для перехода из i -й в $(i+1)$ -ю систему координат, $\mathbf{d}_{i+1/i}^i$ – вектор от центра $i+1$ -й системы координат до центра i -й системы координат, \mathbf{d}_{i/i_c}^i – вектор от центра i -ой системы координат ($i+1$ -ое соединение звеньев) до центра масс звена i .

Каждое i -е звено манипулятора представляет собой цилиндр, на которое действуют сила \mathbf{F}_i и момент T_i [16], имеющие следующий вид:

$$\mathbf{F}_i = \mathbf{M}_i (\mathbf{a}_i^i + \dot{\omega} \mathbf{S} \mathbf{d}_{i/i_c}^i - (\omega_i^i)^2 \mathbf{d}_{i/i_c}^i),$$

$$T_i = I_i \dot{\omega}_i^i,$$

где \mathbf{M}_i – матрица инерции, также содержащая присоединенные массы, I_i – момент инерции (включая присоединенный момент инерции).

Далее рассмотрим силы \mathbf{f}_i и моменты t_i , действующие со стороны вязкой среды на каждое из звеньев манипулятора (также учитывающие взаимное влияние звеньев друг на друга):

$$\mathbf{f}_i = \mathbf{R}_i^{i+1} \mathbf{f}_{i+1} + \mathbf{F}_i - m_i \mathbf{g}_i + \mathbf{b}_i + \mathbf{p}_i,$$

$$t_i = t_{i+1} + (\mathbf{S} \mathbf{d}_{i/i+1})^T \mathbf{R}_i^{i+1} \mathbf{f}_{i+1} + (\mathbf{S} \mathbf{d}_{i/i_c})^T \mathbf{F}_i + T_i + (\mathbf{S} \mathbf{d}_{i/i_c})^T (-m_i \mathbf{g}_i + \mathbf{p}_i) + (\mathbf{S} \mathbf{d}_{i/i_b})^T \mathbf{b}_i + n_i,$$

где \mathbf{g}_i – вектор гравитационных сил, m_i – масса звена манипулятора, \mathbf{b}_i – вектор силы плавучести, \mathbf{p}_i – вектор гидродинамических сил, n_i – момент, создающийся сопротивлением жидкости.

Таким образом, уравнения динамики подводного робота-манипулятора, записанные в связанной с телом системе координат, имеют следующий вид:

$$\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\zeta}} + \mathbf{C}(\mathbf{q}, \boldsymbol{\zeta})\boldsymbol{\zeta} + \mathbf{D}(\mathbf{q}, \boldsymbol{\zeta})\boldsymbol{\zeta} + \mathbf{g}(\mathbf{q}, \boldsymbol{\eta}) = \boldsymbol{\tau}, \quad (1.4)$$

где $\boldsymbol{\zeta} = [\mathbf{v}^T, \dot{\mathbf{q}}^T]^T$ – скорость изменения НПА по x , y , скорость изменения угла дифферента θ , скорость изменения обобщенных координат манипулятора \dot{q}_1 и \dot{q}_2 , $\mathbf{M}(\mathbf{q})$ – матрица инерции, содержащая присоединенные массы, $\mathbf{C}(\mathbf{q}, \boldsymbol{\zeta})$ – матрица кориолисовых и центробежных сил, $\mathbf{D}(\mathbf{q}, \boldsymbol{\zeta})$ – матрица демпфирующих гидродинамических сил, $\mathbf{g}(\mathbf{q}, \boldsymbol{\eta})$ – вектор гравитационных, архимедовых и восстанавливающих сил, $\boldsymbol{\tau}$ – вектор возмущающих и управляющих воздействий.

Стоит отметить, что вектор состояния модели не подлежит полному измерению: вектор скоростей \mathbf{v} оценивается с помощью построенного оптимального асимптотического наблюдателя. Данное обстоятельство объясняется большой технической сложностью измерения скорости [2].

1.2. Линеаризация уравнений динамики

Приведенные уравнения динамики подводного робота-манипулятора являются нелинейными, а методы исследования нелинейных систем очень сложны и не универсальны, поэтому для анализа поведения динамических объектов, а также для синтеза стабилизирующих законов управления используют их линеаризованные модели, свойства и особенности которых подробно изучены.

Рассматриваемый объект управления может быть представлен математической моделью в виде системы нелинейных обыкновенных дифференциальных уравнений:

$$\bar{\mathbf{x}} = \mathbf{F}(\bar{\mathbf{x}}, \bar{\mathbf{u}}), \quad (1.5)$$

где $\bar{\mathbf{x}} \in E^n$ - вектор состояния, $\bar{\mathbf{u}} \in E^m$ - вектор управляющих воздействий и внешнего возмущения, \mathbf{F} - векторная функция, непрерывно дифференцируемая по совокупности своих аргументов.

Выберем некоторым образом векторную функцию, $\bar{\mathbf{u}} = \bar{\mathbf{u}}_p(t)$ и будем считать, что заданы начальные условия $\bar{\mathbf{x}}(0) = \bar{\mathbf{x}}_0$. Тогда в соответствии со свойствами правой части уравнения имеем единственное решение $\bar{\mathbf{x}} = \bar{\mathbf{x}}_p(t)$ задачи Коши для системы (1.5). Таким образом, совокупность указанных векторных функций $\bar{\mathbf{x}} = \bar{\mathbf{x}}_p(t)$ и $\bar{\mathbf{u}} = \bar{\mathbf{u}}_p(t)$ определяют контролируемое движение объекта:

$$\bar{\mathbf{x}}_p = \mathbf{F}(\bar{\mathbf{x}}_p, \bar{\mathbf{u}}_p). \quad (1.6)$$

Рассмотрим отклонения от контролируемого движения:

$$\begin{aligned} \mathbf{x}(t) &= \bar{\mathbf{x}}(t) - \bar{\mathbf{x}}_p(t), \\ \mathbf{u}(t) &= \bar{\mathbf{u}}(t) - \bar{\mathbf{u}}_p(t). \end{aligned} \quad (1.7)$$

Выразим $\bar{\mathbf{x}}(t)$, $\bar{\mathbf{y}}(t)$ из уравнений (1.7) и подставим в (1.5), тогда с учетом тождества (1.6) получаем уравнения в отклонениях от контролируемого движения:

$$\dot{\mathbf{x}} = \mathbf{G}(\mathbf{x}, \mathbf{u}), \quad (1.8)$$

$$\mathbf{G}(\mathbf{x}, \mathbf{u}) = \mathbf{F}(\mathbf{x}_p + \mathbf{x}, \mathbf{u}_p + \mathbf{u}) - \mathbf{F}(\mathbf{x}_p, \mathbf{u}_p).$$

Учитывая приведенные соотношения, получаем что при условиях $\mathbf{x}(t) \equiv \mathbf{0}$ и $\mathbf{u}(t) \equiv \mathbf{0}$ система (1.8) находится в положении равновесия, что соответствует контролируемому движению объекта управления, которое удовлетворяет системе (1.5).

С учетом свойств функций \mathbf{F} , \mathbf{u}_p систему (1.8) можно линеаризовать в окрестности нулевого положения равновесия. В результате имеем линейное уравнение:

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u}, \quad (1.9)$$

где

$$\mathbf{A}(t) = \left\{ \frac{\partial \mathbf{G}_i}{\partial x_j} \right\}_{\mathbf{x}=0, \mathbf{u}=0} = \left\{ \frac{\partial \mathbf{F}_i}{\partial x_j} \right\}_{\bar{\mathbf{x}}=\bar{\mathbf{x}}_p, \bar{\mathbf{u}}=\bar{\mathbf{u}}_p}, i, j = \overline{1, n}; \quad (1.10)$$

$$\mathbf{B}(t) = \left\{ \frac{\partial \mathbf{G}_i}{\partial u_j} \right\}_{\mathbf{x}=0, \mathbf{u}=0} = \left\{ \frac{\partial \mathbf{F}_i}{\partial u_j} \right\}_{\bar{\mathbf{x}}=\bar{\mathbf{x}}_p, \bar{\mathbf{u}}=\bar{\mathbf{u}}_p}, i = \overline{1, n}, j = \overline{1, m}.$$

Как было отмечено выше, в рассматриваемой задаче вектор состояния системы не подлежит непосредственному измерению полностью, поэтому к системе уравнений (1.9) добавим уравнение измерения:

$$\mathbf{y} = \mathbf{C}(t)\mathbf{x}. \quad (1.11)$$

Так как в рассматриваемой задаче частные производные в (1.10) и контролируемое движение не зависят от t , то матрицы \mathbf{A} , \mathbf{B} , \mathbf{C} в уравнениях (1.9), (1.11) константны. Таким образом, линеаризованная модель объекта является стационарной (LTI-системой) и имеет вид:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}\mathbf{x}. \end{aligned} \quad (1.12)$$

Глава 2. Управление роботом-манипулятором в вертикальной плоскости

В настоящей главе рассматривается вопрос построения оптимального асимптотического наблюдателя (фильтра Калмана), а также синтез закона управления посредством решения задачи линейной интегральной квадратичной (LQR) оптимизации.

2.1. Построение оптимального асимптотического наблюдателя

Как было отмечено выше, в рассматриваемой модели в канале измерения происходит потеря информации, так как размерность измеряемого сигнала меньше размерности вектора состояния, более того, измеряемый сигнал обычно зашумлен помехами различного рода. По этой причине один из распространенных аналитических подходов к синтезу законов управления состоит в том, что на базе получаемого измеренного сигнала $y(t)$ модели (2) по некоторым правилам предварительно формируется некоторое приближение $z(t)$ к вектору состояния. Затем происходит обработка полученного вектора оценок для формирования управляющего сигнала. Построение указанных оценок осуществляется с помощью специальных ЛТИ-систем (Linear Time Invariant), называемых асимптотическими наблюдателями [1,2].

Одним из самых часто используемых подходов к построению наблюдателей является наблюдающее устройство Калмана (фильтр Калмана). При этом математическая модель объекта управления дополняется аддитивными стохастическими воздействиями, т.е. возмущениями, носящими случайный характер. При этом математическая модель объекта принимает вид:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} + \mathbf{Gw}(t), \\ \mathbf{y} &= \mathbf{Cx} + \mathbf{Du} + \mathbf{v}(t),\end{aligned}\tag{2.1}$$

где переменная \mathbf{w} представляет собой случайное стационарное возмущение, а \mathbf{v} – случайный стационарный шум (ошибку) в измерениях.

LTI-систему с математической моделью

$$\dot{\mathbf{z}} = \mathbf{Az} + \mathbf{Bu} + \mathbf{F}(\mathbf{y} - \mathbf{Cz} - \mathbf{Du})\tag{2.2}$$

называют асимптотическим наблюдателем Калмана (фильтром Калмана) по отношению к объекту (2.1), если матрица \mathbf{F} при невязках в (2.2) выбрана таким образом, что она является решением оптимизационной задачи о поиске минимума функционала, характеризующего качество процесса наблюдения. Подробно построение фильтра Калмана описано в работе [2].

2.2. Построение стабилизирующего закона управления

Так как вектор состояния модели оценивается с помощью оптимального асимптотического наблюдателя, управление объектом можно осуществить с помощью идеологии LQR-оптимизации. При таком подходе гарантируется асимптотическая устойчивость замкнутой системы.

Математическую модель объекта будем считать заданной в следующем виде:

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}, \mathbf{x} \in E^n, \mathbf{u} \in E^m, \mathbf{x}(0) = \mathbf{d}, t \in [0, \infty).$$

Здесь \mathbf{A} , \mathbf{B} – матрицы с постоянными компонентами, при этом пара (\mathbf{A}, \mathbf{B}) считается стабилизируемой, \mathbf{d} – заданный постоянный вектор. Вместе с моделью объекта управления будем рассматривать уравнение регулятора

$$\mathbf{u} = \mathbf{Kx}\tag{2.3}$$

где \mathbf{K} – постоянная матрица.

Задача LQR-оптимизации состоит в минимизации интегрального квадратичного функционала

$$J = J(\mathbf{K}) = \int_0^{\infty} [\mathbf{x}'(t, \mathbf{K}) \mathbf{Q} \mathbf{x}(t, \mathbf{K}) + \mathbf{u}'(t, \mathbf{K}) \mathbf{R} \mathbf{u}(t, \mathbf{K})] dt \quad (2.4)$$

по множеству матриц \mathbf{K} размера $m \times n$ с постоянными вещественными компонентами, для которых матрица $\mathbf{A} + \mathbf{B}\mathbf{K}$ гурвицева.

Необходимо также учесть, что при использовании оптимального линейного регулятора должны выполняться следующие условия: матрица \mathbf{Q} должна быть знакоположительной и симметрической, а матрица \mathbf{R} – положительно определенной и симметрической.

Глава 3. Имитационный моделирующий комплекс в среде Simulink

Существует множество различных методов анализа поведения динамических систем, но в связи с высоким уровнем развития современных вычислительных средств одним из наиболее широко используемых и эффективных является метод компьютерного моделирования. В силу трудоемкости проведения аналитических исследований и часто невозможности проведения натурных экспериментов, данный подход является простым и удобным способом изучения поведения объекта.

3.1. Компьютерная модель подводного робота-манипулятора

Одним из самых распространенных инструментальных средств поддержки моделирования динамических объектов является система MATLAB с подсистемой динамического моделирования Simulink.

Данная математическая среда включает в себя совокупность базовых и дополнительных прикладных программ, предназначенных для решения широкого спектра математических задач вычислительного характера.

В частности, эта среда содержит различные пакеты прикладных программ для решения задач теории управления: в первую очередь – анализа и синтеза динамических систем.

В настоящее время MATLAB является стремительно развивающейся доступной системой с подробной технической документацией и множеством инструментов, облегчающих разработку. Всё это обуславливает широкое использование этой системы в качестве инструмента для проектирования и исследования динамических объектов.

В данной работе построение компьютерной модели подводного робота-манипулятора осуществляется в подсистеме динамического моделирования Simulink. На базе этой среды выполняется моделирование динамической

системы, ее автоматическая линеаризация, а также осуществляется имитационное моделирование динамических процессов и их визуальное отображение.

На рис. 5 изображена блок-схема нелинейной Simulink-модели объекта, описывающей движение робота. Блоки «Control inputs» и «Disturbance» формируют управляющее воздействие и внешние возмущения соответственно. Блок «Underwater Robot-Manipulator» выдает на свой выход вектор текущих значений правой части уравнения (1.4), приведенного к нормальной форме. Далее этот вектор подается на вход интегратора, который формирует вектор состояния рассматриваемого морского объекта. Полученные результаты отображаются в блоке Scope.

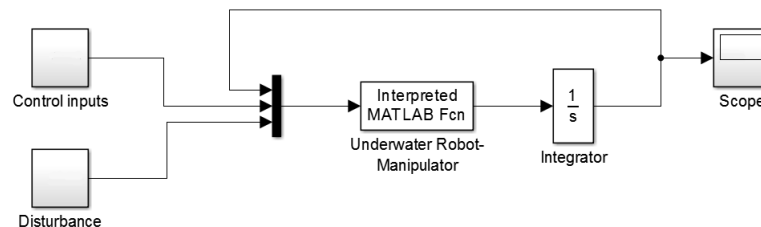


Рис. 5. Simulink-модель подводного робота-манипулятора.

Описанная в главе 1 линеаризация уравнений динамики осуществляется автоматически с помощью функции `linmod()` из пакета MATLAB. Данная функция возвращает линейное приближение модели в SS форме, которая описывается уравнениями (1.11), а на вход подается название Simulink-модели, точка, в окрестности которой нужно произвести линеаризацию, и управляющие силы. Исходный код с вызовом этой функции приведен ниже:

```
% initial conditions
x = [0; 0; 0; 0; 0; x_d; y_d; theta_d; q1_d; q2_d];
u = [0; 0; 0; 0; 0];
sys_lin = linmod('NPSAUV_model_linear', x, u);
```

3.2. Компьютерная модель асимптотического наблюдателя

В среде MATLAB построение наблюдающего устройства Калмана выполняется с помощью функции **kalman()** из пакета Control System Toolbox с использованием следующего исходного кода на языке MATLAB:

```
%матрицы спектральных плотностей возмущения и шума  
Sw = diag([100 100 1000 10 1]);  
Sv = diag([0.1 0.1 0.1*pi/180 0.1*pi/180 0.1*pi/180]);  
  
[Kest, F] = kalman(sys, Sw, Sv);
```

Здесь выходной параметр **Kest** – это LTI-объект, представляющий собой синтезированный фильтр Калмана, а основным выходным параметром является массив **F** – искомая матрица фильтра Калмана.

Результаты работы этой функции используются при построении компьютерной модели асимптотического наблюдателя в системе Simulink (рис. 6).

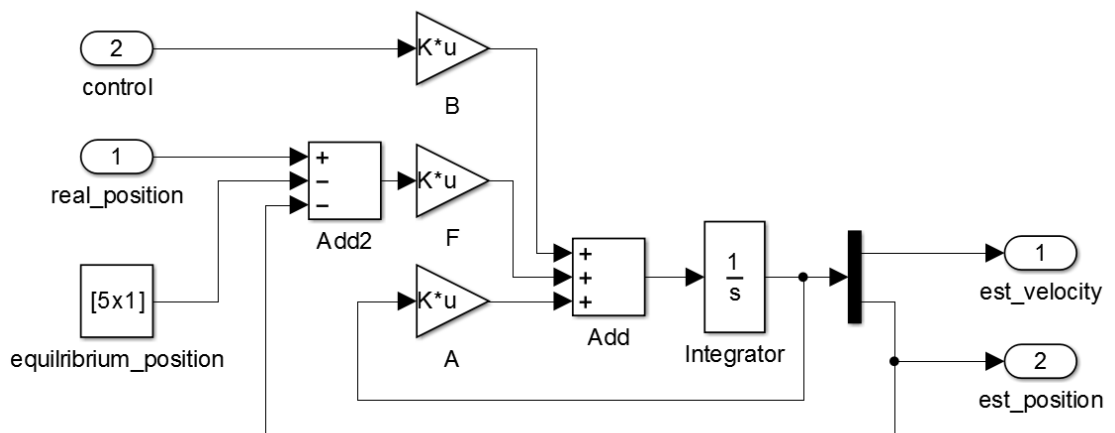


Рис. 6. Simulink-модель асимптотического наблюдателя.

Данная блок-схема представляет собой реализацию уравнения (2.2). Входными сигналами системы являются вектор управления, текущее положение системы, поступающее с измерительных устройств, и положение равновесия, в окрестности которого производится линеаризация уравнений динамики. При этом показания датчиков считаются зашумленными. На

выходе получаем оценку вектора состояния, т.е. оценку скоростей и положения объекта.

3.3. Компьютерная модель системы стабилизирующего управления

Рассмотрим изображенную на рис. 7 компьютерную модель системы стабилизирующего управления, приведенного в главе 2, для подводного робота-манипулятора.

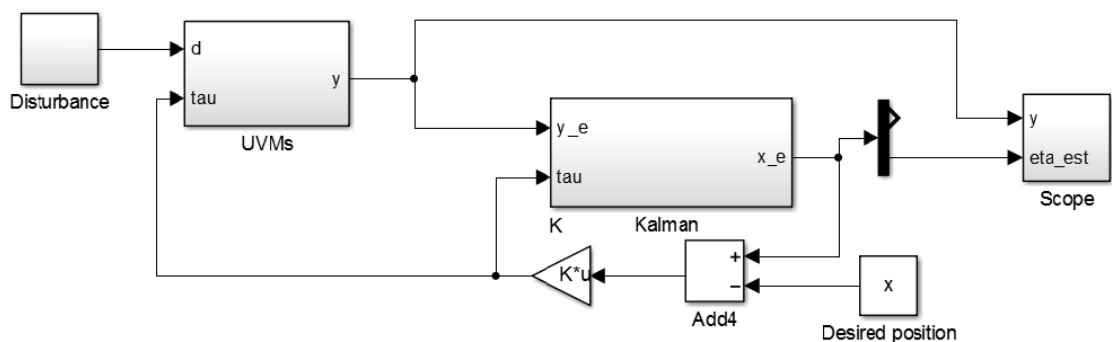


Рис.7. Simulink-модель подводного робота-манипулятора в контуре обратной связи.

Формирование обратной связи происходит следующим образом: оценка вектора состояния модели x_e , получаемая от наблюдателя (блок Kalman) умножается на матрицу с постоянными коэффициентами в блоке K , согласно уравнению (2.3) затем подается на вход системы в качестве управления.

Вычисление постоянной матрицы K уравнения регулятора $u = -Kx$ осуществляется с помощью функции **lqr()** пакета MATLAB. Программный код исполняемого файла приведен ниже:

```
A = sys_lin.a;
B = sys_lin.b;
Q = diag([0.2 0.2 6/pi 6/pi 6/pi 10 10 10 10 10]);
R = diag([0.0001 0.0001 0.000001 1 1]);
[K,S,e] = lqr(A,B,Q,R);
```

На вход функции **lqr()** подаются матрицы A , B линеаризованной системы (1.12) и масштабирующие матрицы Q и R , входящие в функционал

(2.4). На выходе получаем искомую матрицу \mathbf{K} , вектор с корнями характеристического полинома замкнутой системы \mathbf{e} и матрицу \mathbf{S} , являющуюся решением вспомогательного уравнения в задаче оптимизации.

В приложении А приведен полный исходный код описываемых в данной главе функций и подпрограмм.

Глава 4. Результаты имитационного моделирования

Результаты работы оптимального асимптотического наблюдателя и системы управления для подводного робота-манипулятора, представленного уравнением (1.4), могут быть продемонстрированы с помощью имитационного моделирования. Для численного эксперимента используются такие параметры модели, как длина судна (5,3 м) и его масса (5454,54 кг), длины звеньев манипулятора (1 м и 80 см) и их массы (80 и 50 кг).

4.1. Анализ работы асимптотического наблюдателя и системы управления

В рассматриваемой задаче динамического позиционирования подводного робота-манипулятора его желаемое положение задается координатами продольного и вертикального смещения судна, его углом дифферента, а также двумя обобщенными координатами манипулятора. В вычислительных экспериментах желаемое значение координат подводного робота примем следующим: $q_1 = 0,3$ рад, $q_2 = -0,3$ рад, $x = 10$ м, $y = 5$ м, $\theta = 0$ рад.

На рис. 8-12 изображены графики изменения углов звеньев манипулятора и координаты судна, полученные от измерителей, и результат работы фильтра Калмана. В качестве моделей погрешности измерения и стационарного стохастического возмущения используется белый шум.

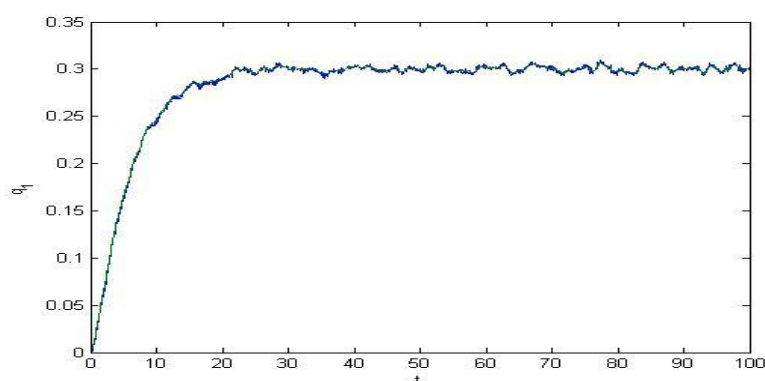


Рис. 8. Измеряемое и наблюдаемое значение q_1 (в радианах).

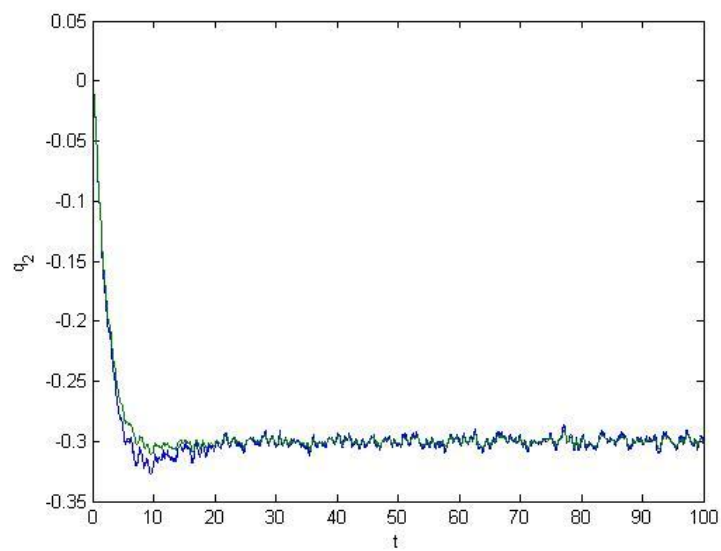


Рис. 9. Измеряемое и наблюдаемое значение q_2 (в радианах).

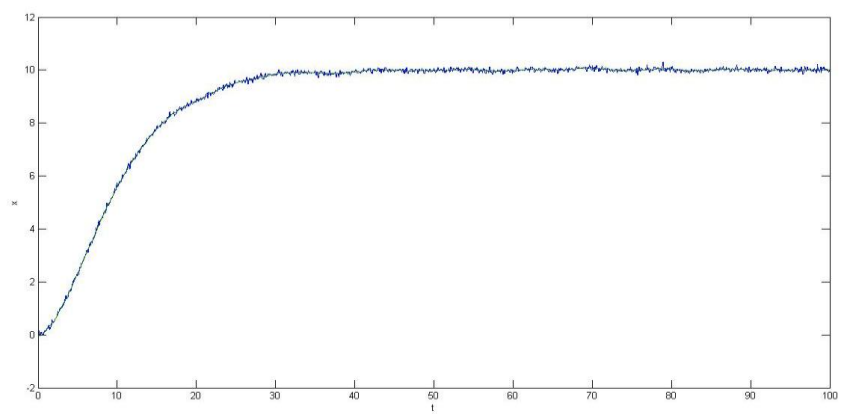


Рис. 10. Измеряемое и наблюдаемое значение x (в метрах).

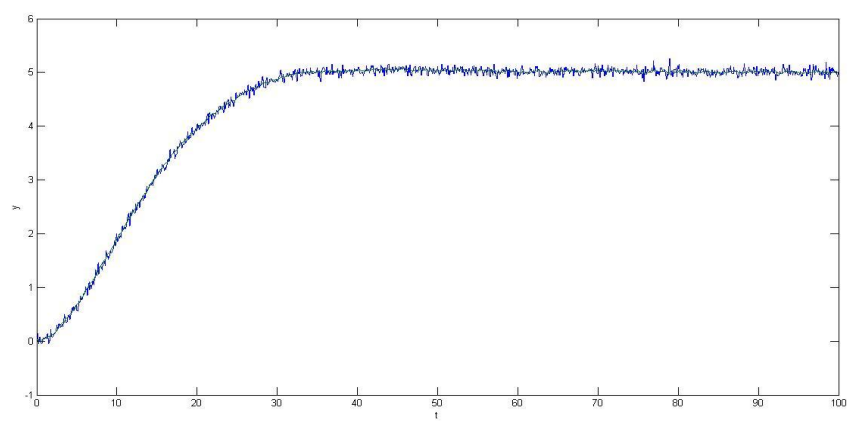


Рис. 11. Измеряемое и наблюдаемое значение y (в метрах).

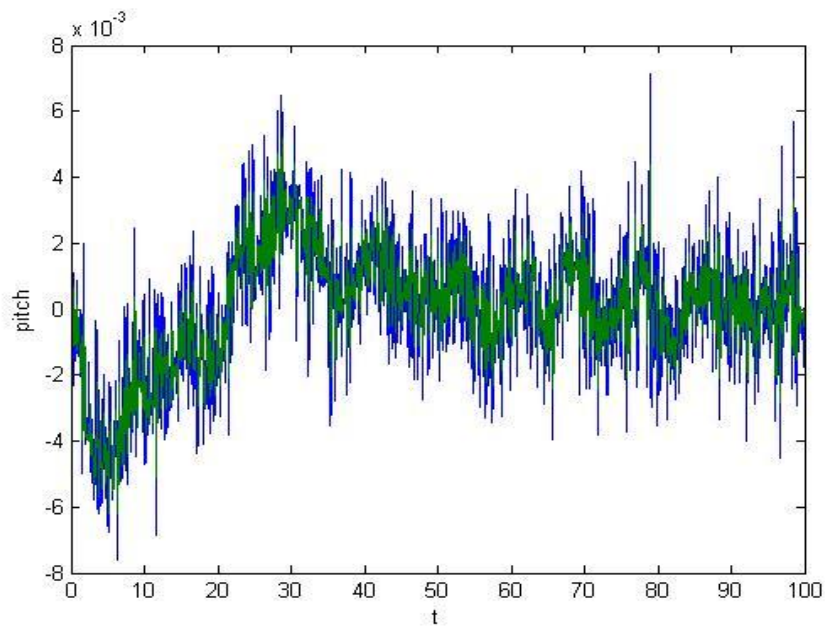


Рис. 12. Измеряемое и наблюдаемое значение θ (в радианах).

Система управления выводит исследуемую модель на заданное положение за 30 секунд. Кроме того, можно заметить, что в данном случае отсутствует перерегулирование, а результат фильтрации близок к реальному значению вектора состояния (рис. 13).

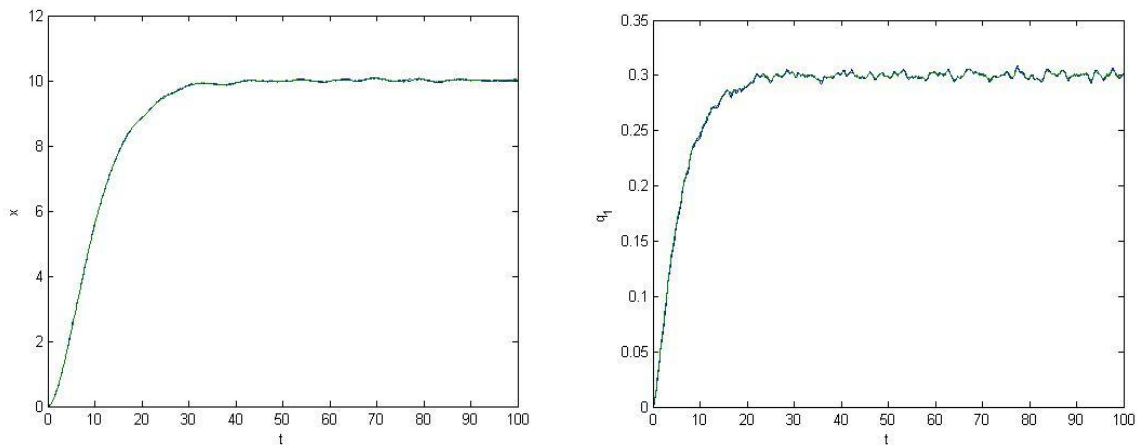


Рис. 13. Фактические значения x и q_1 и результат работы фильтра.

В ходе имитационного моделирования реальные и наблюдаемые обобщенные скорости координат манипулятора \dot{q}_1 и \dot{q}_2 изменялись в соответствии с рис. 14-15.

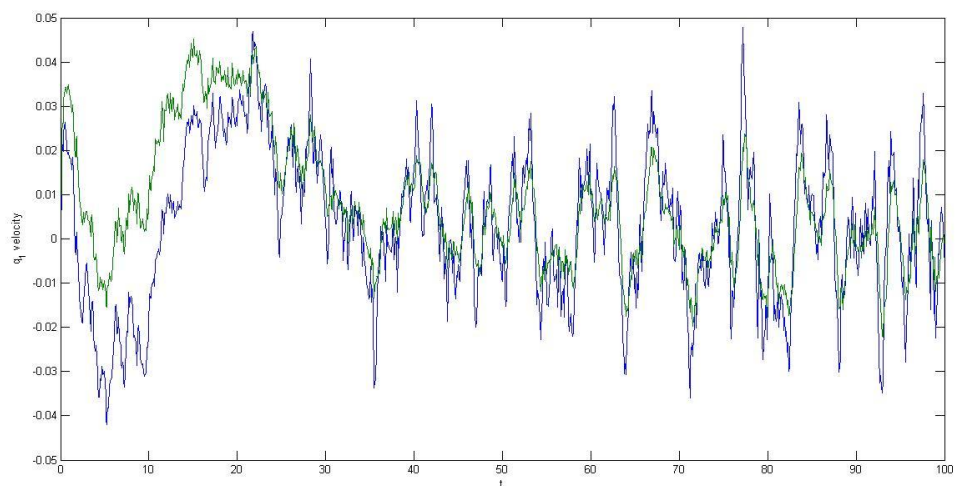


Рис. 14. Фактическое и наблюдаемое значение \dot{q}_1 (в рад/с).

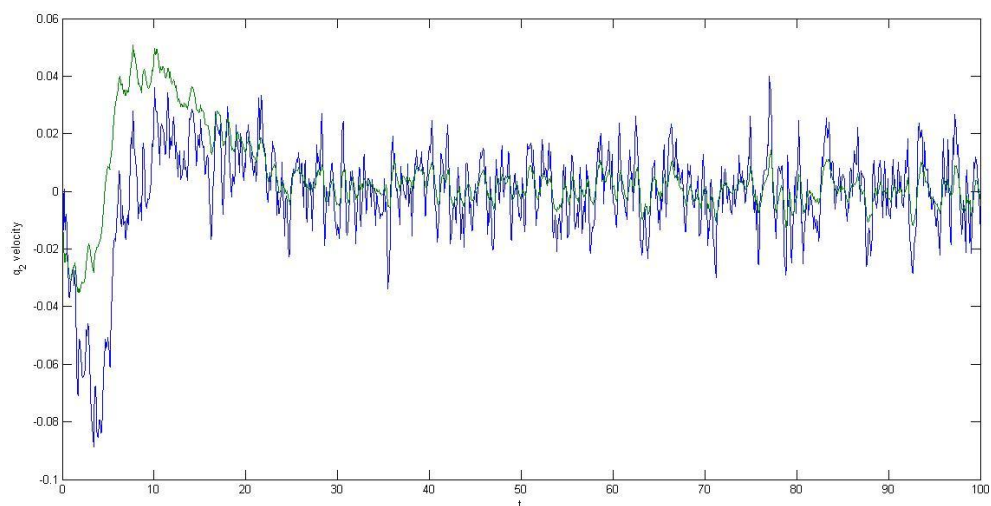


Рис. 15. Фактическое и наблюдаемое значение \dot{q}_2 (в рад/с).

Из графиков видно, что переходный процесс наблюдателя завершается в обоих случаях приблизительно за 20 секунд. После стабилизации положения манипулятора из-за стационарных случайных возмущений внешней среды происходят колебания скоростей, которые не превышают 0,045 и 0,15 рад/с для первого и второго звена соответственно. На рис. 16 приведена ошибка наблюдения координат \dot{q}_1 и \dot{q}_2 .

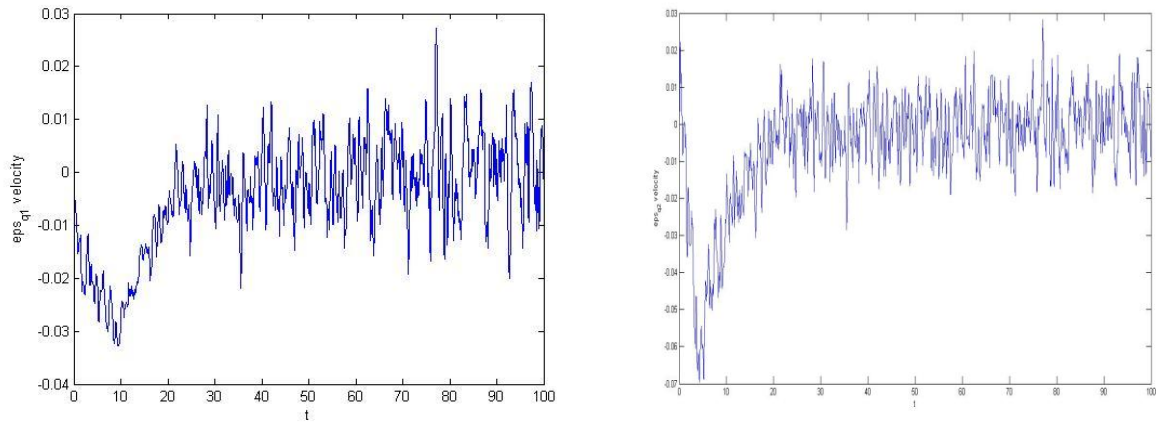


Рис. 16. Ошибка наблюдения координат \dot{q}_1 и \dot{q}_2 (в рад/с).

На рис. 17-19 приведено изменение скоростей координат НПА. Сопоставление приведенных кривых на графиках демонстрирует весьма удовлетворительную работу фильтра Калмана.

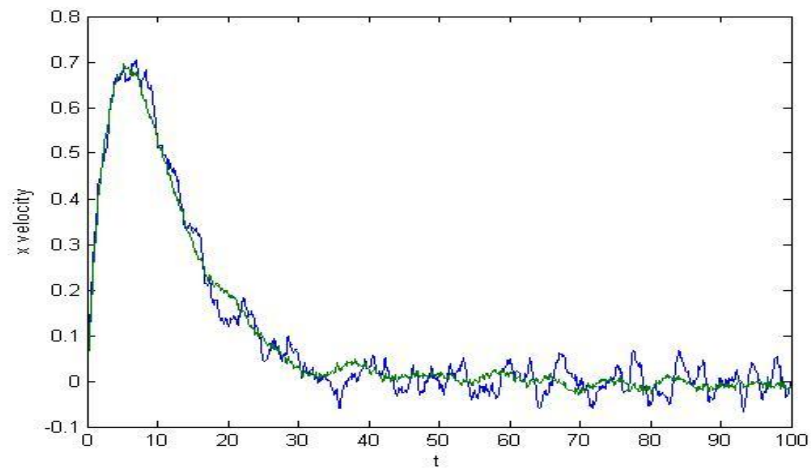


Рис. 17. Фактическое и наблюдаемое значение \dot{x} (в м/с).

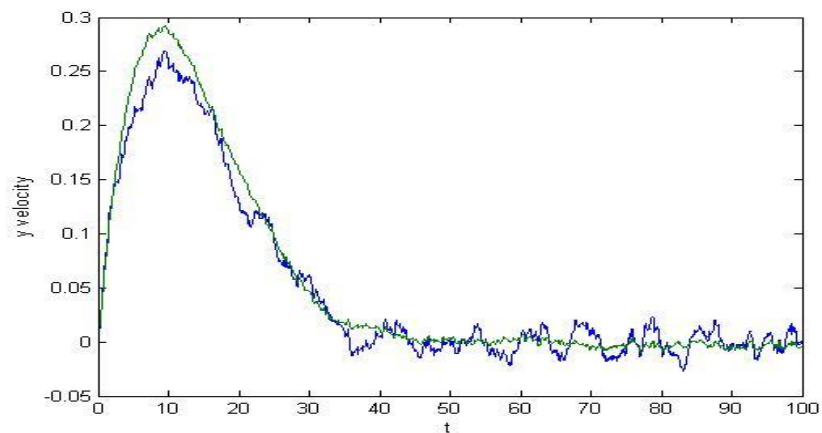


Рис. 18. Фактическое и наблюдаемое значение \dot{y} (в м/с).

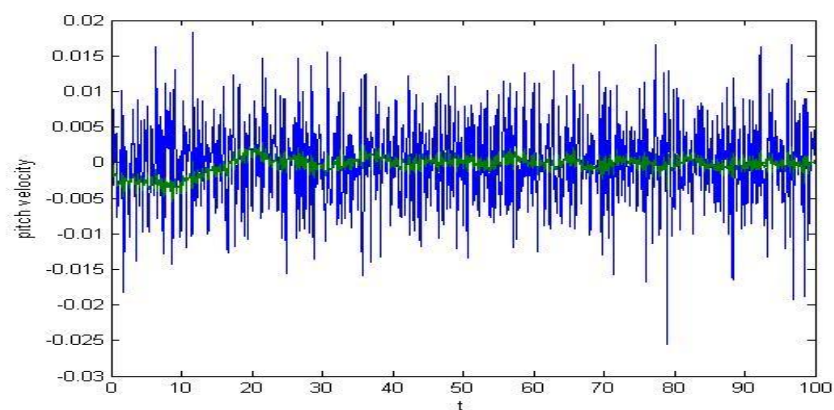


Рис. 19. Фактическое и наблюдаемое значение $\dot{\theta}$ (в м/с).

На рис. 20-21 представлены графики, отражающие погрешность наблюдения.

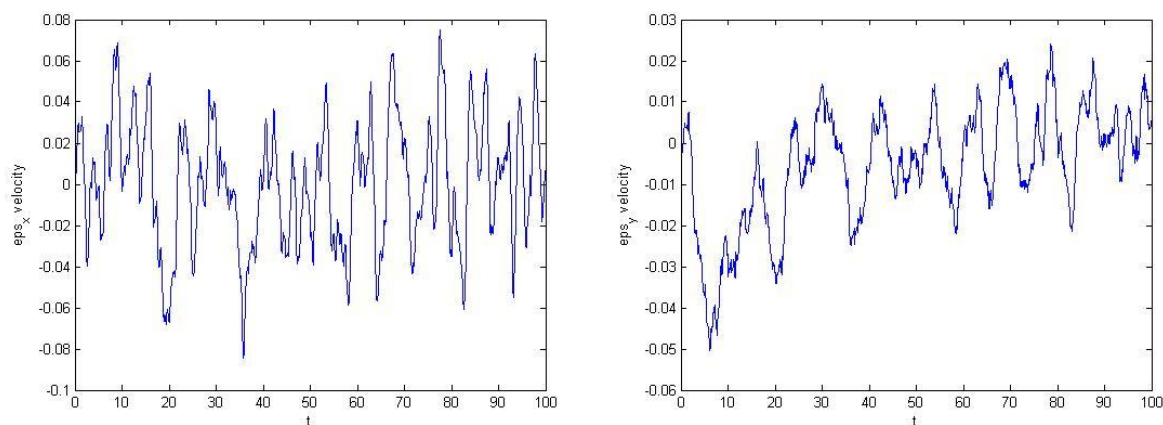


Рис. 20. Ошибка наблюдения координат \dot{x} , \dot{y} (в м/с).

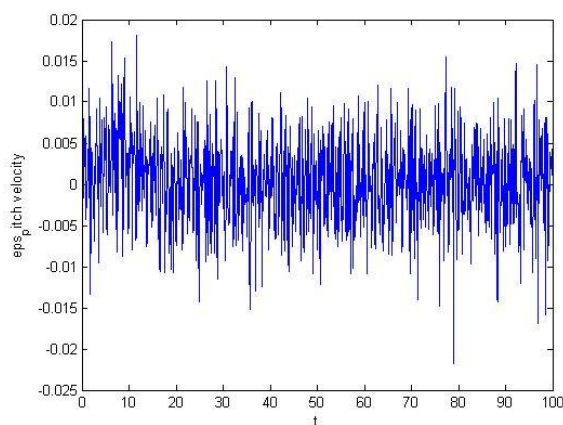


Рис. 21. Ошибка наблюдения координаты $\dot{\theta}$ (в рад/с).

Как видно из графиков, ошибка наблюдения скорости по координате x не превышает 0,08 м/с, по координате y – 0,05 м/с, по углу дифферента θ – 0,02 рад/с.

Выводы

В ходе выполнения работы получены следующие результаты, выносящиеся на защиту:

1. Построена нелинейная математическая модель подводного робота-манипулятора, выполнена ее линеаризация.
2. Сформирован оптимальный асимптотический наблюдатель состояния объекта (фильтр Калмана).
3. Предложен алгоритм стабилизации положения подводного робота с использованием линейной обратной связи.
4. Разработан имитационно-моделирующий комплекс в системе MATLAB-Simulink, и проведено имитационное моделирование поведения замкнутой системы при наличии шумов и помех в канале измерения.

Заключение

В настоящей работе рассмотрена динамическая модель необитаемого подводного аппарата в вертикальной плоскости с установленным на нем манипулятором с двумя степенями свободы, выполнена ее линеаризация. При этом учитывается воздействие моментов и сил со стороны манипулятора на НПА. Для линеаризованной системы построен оптимальный асимптотический наблюдатель, позволяющий получить оценку вектора состояния системы, а также фильтрующий случайный шум в канале измерения и подавляющий стационарные стохастические возмущения.

Кроме того разработан алгоритм управления обратной связью на базе идеологии LQR-оптимизации. Предложенный алгоритм обеспечивает стабилизацию в заданном положении подводного робота. Результаты компьютерного и имитационного моделирования подтверждают эффективность и работоспособность предложенного подхода.

В дальнейшем предполагается совершенствование предложенного алгоритма управления, который бы компенсировал силовые и моментные воздействия манипулятора на судно, обеспечивая, таким образом, астатизм системы управления.

Список литературы

1. Веремей Е.И., Корчанов В.М., Коровкин М.В., Погожев С.В. Компьютерное моделирование систем управления движением морских подвижных объектов. – СПб.: НИИ Химии СПбГУ, 2002. – 370 с.
2. Веремей Е.И. Линейные системы с обратной связью. – СПб.: Лань, 2013. – 448 с.
3. Вукобратович М., Стокич Д., Кирчански Н. Неадаптивное и адаптивное управление манипуляционными роботами. - М.: Мир, 1989.
4. Макаров И.М., Топчеев Ю.И. Робототехника. Истории и перспективы. – М.: Наука; МАИ, 2003. – 349 с.
5. Орлов И.А. Синтез движения манипуляционных систем для пространств со сложными связями и ограничениями: дис. ... канд. ф.-м. наук. Ин-т прикладной математики им. М.В. Келдыша РАН., Москва, 2013.
6. Попов Е.П. Управление роботами-манипуляторами. Изв. АН СССР, Техническая кибернетика, 1974, N 6.
7. Механика промышленных роботов: Учеб. пособие для втузов / под ред. Фролова К.В., Воробьева Е.И. – М.: Высш.шк., 1988.
8. Чернышева И.Н., Мусатов А.К., Глухов Н.А. и др. Силовой расчет, уравнивание, проектирование механизмов и механика манипуляторов: Учебное пособие / Под ред. Мусатова А.К. – М.: Изд-во МГТУ, 1990. – 80с.
9. Antonelli, G. Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems. – Berlin: Springer, 2005. – 265 p.
10. Driankov D., Hellendoorn H. and Reinfrank M. An Introduction to Fuzzy Control. – Berlin: Springer-Verlag, 1995.
11. Fossen T.I. Guidance and Control of Ocean Vehicles. – New York: Wiley, 1994. – 300p.

12. Healey, J. and Lineard D. Multivariable Sliding-Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles // IEEE Journal of Oceanic Engineering, 1993. Vol. 18, No 3, P. 327–338.
13. Janocha, H. and Papadimitriou, I. Simulation of the dynamic behavior of robots in an extreme environment // Robotics and Computer-Integrated Manufacturing, 1991. Vol. 8, No 3, pp. 163-169.
14. Lee M. and Choi H.S. (2000) A Robust Neural Controller for Underwater Robot Manipulators // IEEE Transactions on Neural Networks, 2000. Vol. 11, pp. 1465–1470.
15. Mahesh H. and Yuh J. Control of Underwater Robots in Working Mode // IEEE International Conference on Robotics and Automation, Sacramento, 1991, pp. 2630–2635.
16. Schjolberg I., Fossen T.I. Modeling and Control of Underwater Vehicle-Manipulator Systems // Proceedings the 3'd Conference on Marine Craft Maneuvering and Control (MCMC'94), Southampton, UK, 1994.

Приложение А

В данном приложении приведен полный исходный код функций и вспомогательных программ, встречающихся в работе, на языке MATLAB.

Функция **UVMS** вызывается в блоке Underwater Robot-Manipulator в Simulink-модели, изображенной на рис. 5 главы 3. Функции **Equation**, **kinematics**, **dynamics** и **forces** вызываются функцией **UVMS** и служат для вычисления силовых и моментных воздействий, описанных в главе 1.

```
function [ xdot ] = UVMS( u )

zeta = u(1:5);
eta = u(6:10);
ui = u(11:15);
externalForces = u(16:20);
x = [zeta(1:3);eta(1:3)];

% Dimensional states
u = x(1); v = 0; w = x(2);
p = 0; q = x(3); r = 0;
phi = 0; theta = x(6); psi = 0;

% Rudder and propeller
max_ui(1) = 20*pi/180; max_ui(2) = 20*pi/180;
max_ui(3) = 20*pi/180; max_ui(4) = 20*pi/180;
max_ui(5) = 20*pi/180;

% Parameters, hydrodynamic derivatives and main dimensions
c1 = cos(phi);c2 = cos(theta);c3 = cos(psi);
s1 = sin(phi);s2 = sin(theta);s3 = sin(psi);

L = 5.3;
xG = 0; yG = 0; zG = 0.061;
xB = 0; zB = 0;
rho = 1000; m = 5454.54/(rho/2*L^3);
W = 53400; B = 53400;
Ix = 2038; Iz = 13587;
Ixy = -13.58; Iyz = -13.58; Ixz = -13.58;
Cy = 0; Cz = 0;
Cm = 0; Cn = 0;

r2 = rho*L^2/2;
r3 = rho*L^3/2;
r4 = rho*L^4/2;
r5 = rho*L^5/2;
```

```

Xpp = 7.0e-3; Xqq = -1.5e-2; Xrr = 4.0e-3; Xpr = 7.5e-4;
Xwq = -2.0e-1; Xvp = -3.0e-3; Xvr = 2.0e-2;
Xvv = 5.3e-2; Xww = 1.7e-1;

Zpp = 1.3e-4; Zpr = 6.7e-3; Zrr = -7.4e-3;
Zq = -1.4e-1; Zvp = -4.8e-2; Zvr = 4.5e-2;
Zw = -3.0e-1; Zvv = -6.8e-2;

Mpp = 5.3e-5; Mpr = 5.0e-3; Mrr = 2.9e-3;
Muq = -6.8e-2; Mvp = 1.2e-3; Mvr = 1.7e-2;
Muv = 1.0e-1; Mvv = -2.6e-2;

tau1 = ui(1); tau3 = ui(2); tau5 = ui(3);

% Drag forces and moments
dxL = L/10;
xL = 0;
Ucf = sqrt((v+xL*r)^2+(w-xL*q)^2);

if ~(Ucf == 0),
    for xL = 0:dxL:L
        Ucf = sqrt((v+xL*r)^2+(w-xL*q)^2);
        temp = (0.5*0.6*(v+xL*r)^2+0.6*(w-xL*q)^2)*(v+xL*r)/Ucf;
        Cy = Cy + dxL*temp;
    end

    for xL = 0:dxL:L
        Ucf = sqrt((v+xL*r)^2+(w-xL*q)^2);
        temp = (0.5*0.6*(v+xL*r)^2+0.6*(w-xL*q)^2)*(w-xL*q)/Ucf;
        Cz = Cz + dxL*temp;
    end

    for xL = 0:dxL:L
        Ucf = sqrt((v+xL*r)^2+(w-xL*q)^2);
        temp = (0.5*0.6*(v+xL*r)^2+0.6*(w-
xL*q)^2)*(w+xL*q)/Ucf*xL;
        Cm = Cm + dxL*temp;
    end

    for xL = 0:dxL:L
        Ucf = sqrt((v+xL*r)^2+(w-xL*q)^2);
        temp = (0.5*0.6*(v+xL*r)^2+0.6*(w-
xL*q)^2)*(v+xL*r)/Ucf*xL;
        Cn = Cn + dxL*temp;
    end
end

% Hydrodynamic forces and moments
X = r3*((m+Xvr)*v*r + (Xwq-m)*w*q + Xvp*v*p) + ...
    r4*((m*xG/L+Xqq)*q^2 + (m*xG/L+Xrr)*r^2 - m*yG/L*p*q + ...
    (Xpr-m*zG/L)*p*r + Xpp*p^2) + ...

```

```

r2*(Xvv*v^2 + Xww*w^2 )- (W - B)*sin(theta) + tau1;

Z = r2*(Zw*w*u + Zvv*v^2) +...
    r3*(Zq*u*q + Zvp*v*p + Zvr*v*r) +...
    r4*(Zpp*p^2 + Zpr*p*r + Zrr*r^2) + ...
    (W-B)*cos(theta)*cos(phi) - ...
    m*(rho/2*L^3)*(v*p - u*q + xG*p*r + yG*q*r -
zG*(p^2+q^2))+tau3+ rho/2*Cz;

M = r3*(Muv*u*w + Mvv*v^2) +...
    r4*(Muq*u*q + Mvp*v*p + Mvr*v*r) +...
    r5*(Mpp*p^2 + Mpr*p*r + Mrr*r^2) - ...
    (xG*W-xB*B)*c1*c2 - (zG*W-zB*B)*s2 + ...
    (Iz-Ix)*p*r + Ixy*q*r - Iyz*p*q - (p^2-r^2)*Ixz + ...
    m*(rho/2*L^3)*(xG*(v*p-u*q) - zG*(w*q-v*r)) + tau5 -
rho/2*Cm;

Mv = [6.019e+3    -1.180e-2    3.325e+2 ;
      -1.180e-2    2.332e+4    2.683e+3 ;
      3.325e+2    2.683e+3    4.913e+4 ];

Mv = [ Mv      zeros(3,2);
      zeros(2,3) zeros(2,2)];

%manipulator
Mq = zeros(5,5);
E = eye(5);
zero = zeros(5,1);
g = Equation( eta,zero,zero);

Mq(:,1) = Equation( eta,zero,E(:,1) ) - g;
Mq(:,2) = Equation( eta,zero,E(:,2) ) - g;
Mq(:,3) = Equation( eta,zero,E(:,3) ) - g;
Mq(:,4) = Equation( eta,zero,E(:,4) ) - g;
Mq(:,5) = Equation( eta,zero,E(:,5) ) - g;

Me = Mq-Mq';
Mq = Mq+Mv;

zetadot = Mq\(externalForces +[X;Z;M;0;0]- Equation(
eta,zeta,zero )+[0;0;0;ui(4);ui(5)]);
etadot = [c3*c2*u + (c3*s2*s1-s3*c1)*v + (s3*s1+c3*c1*s2)*w
          -s2*u + c2*s1*v + c1*c2*w
          q
          zeta(4:5)];

xdot = [zetadot;etadot;tau1;tau3;tau5];
end

function [ tau ] = Equation( eta,zeta,zetadot )
    %state vector

```

```

ang = eta(3);
q = eta(4:5);
wB = zeta(3);
eB = zetadot(3);
vB = zeta(1:2);
aB = zetadot(1:2);

qdot = zeta(4:5);
qdotdot = zetadot(4:5);

J = @(phi)[ cos(phi) sin(phi)
            -sin(phi) cos(phi)];
%rotation matrix
RIB = J(ang);
RB0 = eye(2);
R01 = J(q(1))';
R12 = J(q(2))';
% manipulator's parameters
m1 = 80;m2 = 30;rho = 1000;
Cd = 0.6; Ds = 0.4;
L1 = 1;L2 = 0.2;
alfa1 = 0.61; alfa2 = 0.11;
r1 = 0.1;r2 = 0.15;
MAdiag1 = AddedMass( m1,rho,r1,L1 );
MAdiag2 = AddedMass( m2,rho,r2,L2 );
I1z = 80; I2z = 2;

dB0 = [0;0.381];
d01 = [1;0]*alfa1;
d12 = [1;0]*alfa2;
S2 = [0 1;-1 0];

d11c = -d01/2;
d22c = -d12/2;

d01c = d01+d11c;
d12c = d12+d22c;
RBI = RIB';
R0B = RB0';
R10 = R01';
R21 = R12';
[ w0,e0,v0,v00c,a0 ] = kinematics( wB,eB,vB,aB,0,0,R0B,dB0,
-dB0 );
[ w1,e1,v1,v11c,a1 ] = kinematics(
w0,e0,v0,a0,qdot(1),qdotdot(1),R10,d01,d01c );
[ w2,e2,v2,v22c,a2 ] = kinematics(
w1,e1,v1,a1,qdot(2),qdotdot(2),R21,d12,d12c );

M1 = diag(m1*ones(2,1)+MAdiag1(1:2));
M2 = diag(m2*ones(2,1)+MAdiag2(1:2));

```

```

I1 = I1z+MAdiag1(3);
I2 = I2z+MAdiag2(3);

[ F1,T1 ] = forces( w1,e1,a1,M1,I1,d11c );
[ F2,T2 ] = forces( w2,e2,a2,M2,I2,d22c );

R1I = R10*R0B*RBI;
R2I = R21*R1I;

g = [0;9.81];

mg1 = m1*R1I*g;
mg2 = m2*R2I*g;
b1 = R1I*MAdiag1(2)*g;
b2 = R2I*MAdiag2(2)*g;
p1 = 2*Ds*v11c + rho*Cd*2*r1*L1*norm(v11c)*v11c;
p2 = 2*Ds*v22c + rho*Cd*2*r2*L2*norm(v22c)*v22c;
n1 = v1'*S2*(MAdiag1(1:2).*v1);
n2 = v2'*S2*(MAdiag2(1:2).*v2);

d01b = d01/2;
d12b = d12/2;
[f2,t2] = dynamics(
[0;0],0,F2,T2,zeros(2,2),d12,d12c,d12b,mg2,b2,p2,n2 );
[f1,t1] = dynamics(
f2,t2,F1,T1,R12,d01,d01c,d01b,mg1,b1,p1,n1 );

tau = [RB0*R01*f1;t1 + [-dB0(2),dB0(1)]*R01*f1;t1;t2]
end

function [ wkp1,ekp1,vkp1,vkkc,akp1 ] = kinematics(
wk,ek,vk,ak,qkp1dot,qkp1dotdot,Rkp1k,dkkp1,dkkp1c )
    wkp1 = wk + qkp1dot;
    ekp1 = ek + qkp1dotdot;
    vkp1 = Rkp1k*vk + wkp1*[-dkkp1(2);dkkp1(1)];
    vkkc = Rkp1k*vk + wkp1*[-dkkp1c(2);dkkp1c(1)];
    akp1 = Rkp1k*ak + ekp1*[-dkkp1(2);dkkp1(1)] -
wkp1*wkp1*dkkp1;
end

function [ fk,tk ] = dynamics(
fkp1,tkp1,Fk,Tk,Rkkp1,dkm1k,dkm1kc,dkm1kb,mgk,bk,pk,nk )
    fk = Rkkp1*fkp1 + Fk + pk;
    tk = tkp1 + [-dkm1k(2), dkm1k(1)]*Rkkp1*fkp1+...
[-dkm1kc(2), dkm1kc(1)]*Fk + Tk +...
[-dkm1kb(2), dkm1kb(1)]*(pk)+...
nk;
end

function [ Fk,Tk ] = forces( wk,ek,ak,Mk,Ik,dkkc )

```

```

    Fk = Mk*(ak + ek*[-dkkc(2);dkkc(1)] - wk*wk*dkkc);
    Tk = Ik*ek;
end

```

С помощью следующего программного кода выполняется линеаризация системы, происходит настройка оптимального асимптотического наблюдателя (фильтра Калмана), а также формируется обратная связь:

```

%linearization
x = [0;0;0;0;0;10; 5; 0; 0.3; -0.3];
u = [0; 0; 0; 0; 0];
sys_lin = linmod('NPSAUV_model_linear', x, u);

G = [eye(5); zeros(5)];

%observer
Ak = sys_lin.a;
Bk = [sys_lin.b G];
Ck = sys_lin.c;
Dk = [sys_lin.d zeros(5)];
sys = ss(Ak, Bk, Ck, Dk);

Sw = diag([1 0.1 10 1 10]);
Sv = diag([0.1 0.1 0.1*pi/180 0.1*pi/180 0.1*pi/180]);
[Kest, F, Y] = kalman(sys, Sw, Sv);

%control law
A = sys_lin.a;
B = sys_lin.b;

Q = diag(1./[0.7 0.7 0.05 0.1 0.2 5 5 pi/6 pi/6 pi/6 ].^2);
R = diag(1./[1000 2000 2000 20 3].^2);
N = zeros(10,5);
[K,S,e] = lqr(A,B,Q,R,N);

```